

UNITED STATES APPLICATION

OF

AWELE NDILI, SCOTT MOELLER

FOR

**SYSTEM FOR AUTOMATIC DATA RETRIEVAL ON AN INTERNET
PROTOCOL NETWORK**

PREPARED BY WILSON SONSINI GOODRICH & ROSATI

BACKGROUND OF THE INVENTION

Priority Information

This application claims priority to provisional U.S. Patent App. No.
5 60/163,115, entitled "Portal Configuration in Wireless Medium", naming Scott
Moeller and Awele Ndili as inventors, filed on November 2, 1999, and
incorporated by reference herein.

Field of the Invention

This invention relates to the field of network interfaces and systems
10 operating under the Internet Protocol. In particular, the invention relates to an
interface to automatically access events on a broadband network for a selected
network enabled device.

Description of the Related Art

Wireless devices such as cell phones have limited band width and data
15 entry capacity. The limited bandwidth significantly reduces the number,
quantity and quality, of web-sites that can be made available to the user. As a
result, users of wireless mediums such as cell-phone networks often have
limited choices in which web-sites they can visit, and also experience lengthy
download times.

20 Wireless devices are also very compact, with small and minimal set
keypads, making data entry into the device difficult to the user in comparison to
using traditional computer devices such as laptops and desktop computers. As

such, entering data to visit web-sites is more difficult with current wireless devices such as cell-phones.

Further, e-commerce sites require visitors to respond to queries for personal information. Personal information includes name, billing address, shipping address, credit card number and items purchased. These queries require users to respond with multiple alpha-numeric type entries.

Electronic messaging services are becoming more prevalent on networks. Many individuals have two or more messaging accounts, requiring users to continuously access different messaging services to retrieve emails.

10

SUMMARY OF THE INVENTION

An object of the invention is to provide an automatic system that notifies users through an end device of an electronic message or other type of network event.

15

Another object of the invention is to provide a system to push network events to wireless end devices, such as cell-phones, pagers and PCS devices.

Another object of the invention is to provide a unified messaging center to monitor multiple messaging accounts.

20

Another object of the invention is to provide a user-configurable system to automatically retrieve messages and other network events from multiple sites and services.

Still, another object of the invention is to provide a user-configurable system to push electronic messages and other network events to end devices

selected by the end user, including wireless devices such as cell-phones and pagers.

005220-1552560

BRIEF DESCRIPTION OF THE FIGURES

FIG. 1 illustrates a block diagram for a network retrieval and notification system, under an embodiment of the invention.

FIG. 2 illustrates a flow chart for a network retrieval and notification
5 system, under an embodiment of the invention.

FIG. 3 illustrates a block diagram illustrating signals conveyed through an exemplary architecture for a message retrieval and notification system, under an embodiment of the invention.

FIG. 4 illustrates a flow chart, for a message retrieval and notification
10 system, under an embodiment of the invention.

FIG. 5 illustrates a flow chart for fetching messages from a web-based messaging service, under an embodiment of the invention.

FIG. 6 illustrates a flow chart for fetching messages from a non-web messaging service, under an embodiment of the invention.

FIG. 7 illustrates a block diagram for a system that retrieves network
15 events for a wireless device, under an embodiment of the invention.

FIG. 8 illustrates a flow process for retrieving messages from different messaging services and pushing the messages to an end device operating under a wireless environment, under an embodiment of the invention.

FIG. 9 illustrates a system to respond to retrieved messages from an end
20 device, under an embodiment of the invention.

FIG. 10 is a block diagram of an alternative embodiment of the invention.

DETAILED DESCRIPTION

A. System Overview

Embodiments of the invention provide an interface between an end device and one or more network sites. The network sites operate under any IP
5 protocol, such as HTML or POP3. The end device includes any network enabled device, such as wireless phones, pagers, and handheld computers.

In one embodiment, the system automatically retrieve emails received in messaging accounts for a wireless end device. The messaging accounts may operate under different protocols. The system then reformats the retrieved
10 emails for a wireless medium. The system is configurable by an end user in retrieving messages and in signaling retrieved messages to the end device.

The system may also retrieve web-based content for the end device. The system is configurable by a user of the end device to retrieve select content from specific web-sites. Embodiments of the invention enable users to
15 configure the system to programmatically communicate with network sites to access or interact with web-based events. Examples of web-based events that can be provided to the user through the end device include e-commerce applications, on-line auctions, stock quotes, travel arrangements, and messaging.

20 In contrast, previous systems allow users of the end device to manually access network sites for emails and/or web content. The inherent limitations of the end device made network access more difficult to the user. Furthermore,

wireless devices are too limited in bandwidth for broadband networks such as the Internet.

Fig. 1 is a block diagram illustrating a system that notifies an end user of a network event. The system includes an engine 50 that communicates with a plurality of sites 10, 20. A user terminal 60 is used to signal user-defined configurations and information to a database of engine 50. The engine 50 accesses the database to fetch selected network accessible resources and events from the plurality of sites 10, 20, using user-defined configurations and other information.

The engine 50 is configurable to notify an end device 70 of the selected information available on the plurality of sites. The engine 50 may also deliver to end device 70 the selected information retrieved from sites 10, 20.

In an embodiment, engine 50 receives the information from the plurality of sites in a format corresponding to a first network protocol. The engine 50 transmits the information to end device 70 in a medium or protocol that is different than the first network protocol. In an embodiment, engine 50 receives the information in any Internet protocol (IP), and signal the information to an end device 70 operating under a wireless protocol.

Embodiments of the invention may be employed with any Internet Protocol (IP)-based network protocol, including Transport Control Protocol (TCP) and UDP. In an Internet space, the network event may be a web-based event. Examples of TCP/IP protocols include HTML, KTP, Telnet, POP3, and Gopher. Other examples of IP protocols include UDP-based protocols. Network

accessible resources and events include files, content, and resources accessible on a network. Network events include electronic messages, as well as web-based events associated with specific sites on the Internet. Electronic messages include emails, instant messages, files existing as attachments to electronic

5 messages, programmatic notifications of events generated by server-side modules of third parties (stock alerts), and multimedia type messages. For example, network events include emails in an HTML or POP3 protocol. Web events maybe associated with an HTML link that accesses the web event. Web events also include text or media resources appearing or linked to web pages.

10 In other embodiments, network events include a series of interactions with server-side modules that are accessible through links. Network interactions may include prompts from server-side modules. For example, e-commerce applications provide access to servers that receive purchasing information for a selected item. Network events also include real-time

15 information appearing on, for example, a web page.

In an embodiment, sites 10, 20 are all web-based sites operating under an HTML protocol. In another embodiment, sites 10, 20 operate under different protocols. For example, site 10 may be a web-based site operating under an HTML protocol, and site 20 may be a messaging system using a POP3 protocol.

20 The end device 70 includes any network enabled device. Preferably, the end device 70 is a wireless device, operating under a wireless access protocol. Examples of end devices 70 include cellular phones operating in a PCS environment, handheld computers including Palm and Windows CE devices,

5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
95
100
105
110
115
120
125
130
135
140
145
150
155
160
165
170
175
180
185
190
195
200
205
210
215
220
225
230
235
240
245
250
255
260
265
270
275
280
285
290
295
300
305
310
315
320
325
330
335
340
345
350
355
360
365
370
375
380
385
390
395
400
405
410
415
420
425
430
435
440
445
450
455
460
465
470
475
480
485
490
495
500
505
510
515
520
525
530
535
540
545
550
555
560
565
570
575
580
585
590
595
600
605
610
615
620
625
630
635
640
645
650
655
660
665
670
675
680
685
690
695
700
705
710
715
720
725
730
735
740
745
750
755
760
765
770
775
780
785
790
795
800
805
810
815
820
825
830
835
840
845
850
855
860
865
870
875
880
885
890
895
900
905
910
915
920
925
930
935
940
945
950
955
960
965
970
975
980
985
990
995

paggers, and desktop computers. Examples of wireless formats and protocols include HDML, WML, WAP.

In an embodiment, engine 50 comprises one or more fetch and notification modules to access events on sites 10, 20. The fetch and notification modules may be made specific to individual network sites. For example, engine 50 may include a plurality of “smart” fetch and notification modules to access different messaging services, including Internet services such as, Hotmail™ or Yahoo™ email. The engine 50 associates each Internet messaging service with a specific fetch and notification module. Each fetch and notification module is programmed to access necessary preliminary web-pages for that specific messaging service. Each fetch and notification module may also be programmed on how user-defined or configured information should be signaled to the service, and which page to parse for links to messages.

In other embodiments, engine 50 includes smart fetch and notification modules that are specific to individual web-sites, or events on individual web-sites. As an example, engine 50 may include fetch and notification modules to periodically retrieve the latest auction information on a particular item, or class of item. The user may visit the auction site, select an item, identify the item’s auction number, and configure engine 50 to periodically access the auction item according to the identification number. The frequency at which the auction item is retrieved by engine 50 is also configurable. The user may, for example, instruct engine 50 to retrieve the last bid, and reserve amount from the auction site at a designated period before the auction is over.

005220-4554-02500

In similar fashion, the fetch and notification modules may identify specific e-commerce sites, stock brokerage sites, travel sites, etc. For example, engine 50 may be programmed to retrieve real-time quotes from a stock site every ten minutes. These web-events are then pushed to end device 70, as discussed below.

The engine 50 is programmable through an editor interface (not shown) to fetch events from selected non-messaging network sites. In an Internet space, non-messaging sites may provide, for example, e-commerce, auctions, weather reports, brokerage services, travel services, online banking etc. The engine 50 includes a smart fetch and notification to retrieve information from individual non-messaging sites.

The engine 50 includes a conversion module to convert retrieved network events from a first protocol to a protocol of end device 70. For example, events under HTML and POP3 protocols may be converted to a wireless protocol before the events are pushed to end device 70. In an embodiment, engine 50 includes multiple conversion modules. Each conversion module is designated to convert events from one kind of protocol to another protocol for end device 70. As with fetch and notification modules, conversion modules are intelligent, i.e. specific to one of the sites 10, 20.

The engine 50 may also include a user-configurable database that stores information to access and/or select information from sites 10, 20. The database may be maintained as an accessible account for the given user. The user account enables users to configure engine 50 to retrieve specified types of network

events from select sites. The user may configure the associated account to cause engine 50 to select fetch and notification modules, as well as conversion modules. The user may also configure the associated account to provide engine 50 with parameters for use with fetch and notification modules, as well as the
5 conversion modules.

In an embodiment, configuration terminal 60 couples to engine 50 through the Internet. The configuration terminal 60 may include any Internet enabled computer system. The user may, for example, access the user's account on engine 50 using a personal desktop or laptop computer. Engine 50 may host
10 a web page to receive user-defined configurations and information.

The types of user-defined configurations information that may be entered and stored on engine 50 include preferences, raw user-data, and directives. The user may set preferences on how often certain sites should be accessed, or how often the user should be notified of a web-event. The user may
15 provide raw data regarding the user, such as billing information for e-commerce applications, and login/password information for messaging services. The user may provide directives on which sites to access. The user may also configure engine 50 to check for messages with different messaging accounts, including messaging accounts on different messaging services.

20 For example, the user may specify for engine 50 to fetch from an auction site items matching a specific criteria every three days. Alternatively, the user may specify addresses to email accounts to the web page, as well as password and login information for each of the specified email accounts. Still further, the

user may also specify credit card information, favorite web-sites, and frequency in which engine 50 should check email accounts or web-sites.

In another embodiment, the user may also configure the user account so that engine 50 automatically retrieves events from programmatically selected web-sites. The events correspond to categories and topics specified by the user. For example, the user may specify types of information that the user prefers to be notified of, such as stock alerts and weather, daily news, etc.

In an embodiment, engine 50 programmatically identifies configurable information entered from configuration terminal 60. Fetch and notification modules are assigned to each user account based on configurations and information provided by the user. The engine 50 then accesses the specified sites for select web-events, as specified by preferences, data, and directives supplied from user terminal 60.

The engine 50 then notifies the end device 70 of the retrieved events. The user may configure engine 50 for a particular end device. The engine 50 applies a conversion module for the selected end device. The engine 50 converts retrieved network events into a format, protocol, and/or medium for the specified end device. For example, the user may access a web-page hosted by engine 50, and specify the type of cell-phone, pager, and /or handheld computer that the user will employ with the system of FIG. 1. Thus, the user may configure engine 50 to retrieve information from select sites, and to signal the information to a particular type of end device 70.

FIG. 2 details a fetch and notification process in which engine 50 notifies a specific user of network events based on configurations and information provided by that user. Reference is made to elements shown in FIG. 1.

5 In step 210, a given user configures the associated account on engine 50. The configurations may specify preferences, raw data, and directions for engine 50. The account may be stored on engine 50. In an embodiment, user terminal 60 includes a user-interface to interact with engine 50. The given user may enter all configurations in one sitting, or modify and add configurations over repeated
10 sittings.

Based on information provided by the given user, the engine 50 accesses the first site 20 in step 220. The first site may operate under a first protocol, such as HTML or POP3. For HTML sites, the first site 20 may have a first domain. The first site may also correspond to a first account (such as a
15 messaging account) on a domain. The first site 20 is selected because of configurations provided by the user in step 210. Further, the first site 20 may be accessed for selected information based on protocols and procedures that are specific to the first site 20.

In step 230, engine 50 identifies selected events that are to be retrieved
20 from first site 20. Certain events may be selected for purposes of making a determination as to whether the user should be notified of the event. For example, messages may be identified as new or old, with the given user being notified of only the new messages. The events may also be selected to match a

type or criteria of the given user's configurations. For example, a signature for each event may be compared with a database of signatures types. If the signature of the event matches the desired signature type, the information is assumed to be of a particular type and selected for retrieval. If the first site 20
5 operates under an HTML protocol, the signature types may be used to identify, for example, e-commerce items (auction items in a specific criteria), stock quote listings, weather information etc.

As shown by an embodiment of FIG. 2, engine 50 accesses a second site 30 in step 240. The second site 30 may operate under a second protocol, such as
10 HTML or POP3. Alternatively, the second site 30 may have a second domain, but the same protocol as the first site 20. Alternatively, second site 30 may correspond to a second account (such as a messaging account) with the same protocol as the first site 20. Still further, the second site may correspond to a second account on the same domain as the first site. Therefore, either the
15 protocol, domain, or account may be different for the second site 30 when compared to the first site 20.

In step 250, the engine 50 identifies selected events on second site 30. The second site 30 may be accessed for selected information in a manner similar to the first site 20. The engine 50 may use configurations provided by
20 the given user for the second site. In addition, the second site 30 may be accessed for selected information based on protocols and procedures that are specific to the second site 30. As previously stated, the protocols to access and

retrieve information from the second site 30 may be different than similar protocols and procedures used in step 220.

Steps 220-250 may be repeated for other network sites. The number of sites accessed for selected events may be determined by information provided by the given user. For example, the given user may configure the given user's account to access and select events from tens or hundreds of web-sites automatically. These sites may include Internet messaging sites, POP3 messaging sites, e-commerce sites etc.

In step 260, selected events retrieved from the first site 20 and second site 30 are pushed to the end device 70. In embodiments of the invention, the events may be converted from one or more protocols to another protocol, in a manner to be discussed in greater detail. The type of end device 70 may be pre-specified, or otherwise provided as configured information, from the user.

B. Messaging Applications

FIG. 3 is a block diagram to illustrate a messaging application, under an embodiment of the invention. In a messaging application, engine 50 fetches events corresponding to electronic messages such as emails from messaging services. While emails may be explicitly mentioned, embodiments described below also include other forms of electronic messages, such as listed elsewhere in this application. The engine 50 can access different messaging services to notify the end device 70 that a new email has arrived.

A plurality of messaging services such as shown by numerals 110, 120 and 130 are accessible to engine 50. The engine 50 is configurable to retrieve

005220-155E1560

emails from messaging services 110-130. In such an embodiment, configuration terminal 60 includes any Internet enabled device. The user accesses engine 50 over the Internet and configures engine 50 to retrieve emails from specified messaging services 110-130. The user also configures engine 50 to push retrieved messages to a specific end device, such as a wireless device. The engine 50 signals emails obtained from messaging services 110-130 to the specified end device 70. In an embodiment, the user accesses engine 50 via the Internet to specify the make, model, and/or type of end device(s) 70.

In an embodiment, engine 50 accesses the messaging services 110-130 to determine if new emails have been received. The engine 50 signals a notification to end device 70 that new emails have arrived to identified messaging accounts on messaging services 110-130. The engine 50 may also signal a content of the new messages to end device 70.

The engine 50 may communicate with end device 70 through a format or protocol conversion. In an embodiment, engine 50 receives new emails from the different messaging services 110-130 in a first format. The first protocol may, for example, be an HTML or POP3 protocol. The engine 50 converts the formatting to another protocol for delivery to end device 70. The end device 70 may be any network enabled device, including computer systems (desktop and laptop computers), personal digital assistants (PDA), pagers and cell-phones. In one embodiment, end device 70 is WAP enabled device. The engine 50 converts the messages from the HTML or POP3 protocol to an HDML format suitable for end device 70.

009220-4551560

The messaging services 110-130 include any system that maintains electronic messaging accounts for users. Further, the messaging services 110, 120 and 130 may operate under one or more different protocols. For example, the first messaging service 110 may operate under a web-based protocol, such as HTML. Examples of such messaging services include web-based email services, such as provided by Yahoo™ mail and Excite™ mail. The second messaging service 120 may operate on another IP protocol such as POP3.

In general, messaging services such as POP3 store emails for a given user's account on a server. The given user accesses the account using a terminal that directly links to the server. To automatically retrieve email, the given user has to designate a receiving terminal that periodically checks the server on the network for new email. In contrast, users of web-based messaging services can use any Internet enabled terminal to link with the messaging service. Information such as password and login information are typically provided to the web-based messaging services to access emails. An Internet messaging service may also access password and login information by identifying the given user through a cookie on the terminal.

The engine 50 includes a database of information used to access multiple messaging services 110-130. The database includes a depository of messaging modules that include information on accessing the multiple messaging services. The messaging modules include, for example, Internet addresses to web-based messaging services 110, and phone numbers or addresses to non-web messaging services 120. The messaging modules further

include the protocol and/or format used by each specific messaging service.

The messaging modules may also include specific procedures on accessing each of the messaging services. For example, each messaging module may cause engine 50 to execute a sequence in which a prompt for a login or password is received, and further cause engine 50 to identify emails stored with the messaging protocol under a particular protocol, such as HTML or POP3.

The email depository is accessible to different accounts associated with users of a system. A given user can configure an account using the configuration terminal 60. The configuration terminal 60 may correspond to, for example, an Internet enabled system that can communicate with engine 50 over the Internet. Each account may be configured for preferences and email directories, as well as other information such as described above. The engine 50 accesses the messaging module associated with each email directory specified by the given user. The engine 50 may access the messaging service periodically, based on a preference specified by the given user's account.

The given user may configure engine 50 to retrieve messages from messaging accounts on one or more messaging services. For example, the given user may identify a personal email account on an Internet email account as the first messaging service 110. The user may identify a work email account on the second messaging service 120 operating under a POP3 protocol. Still further, the user may specify email accounts on a third messaging service operating under another protocol, such as another web-based protocol. For example, the given user may identify a personal email account on America Online™ (AOL)

as the third messaging service 130. The engine 50 identifies a messaging module tailored or specific to the messaging service identified by the given user.

In addition to identifying accounts with different messaging services, the given user may configure the user's account on engine 50 with other preferences,

FIG. 4 illustrates a flow process in which a system operates to retrieve messages from different messaging services, under an embodiment of the invention. In step 410, the given user configures an associated account maintained by engine 50. The configured account identifies email accounts that are accessed by engine 50 for retrieval of emails. The given user may configure the associated account by providing specific information that enables engine 50 to access the email accounts. For web-based messaging services, the given user may provide, for example, an Internet domain where the messaging service resides, as well as login and password information for that account. In addition, the user may specify other "screen-names" or accounts on the same domain or messaging service. For example, messaging services such as Yahoo™ and AOL™ allow users to maintain multiple accounts using a single login and password. The user may specify the different screen-names used with the web-based messaging services.

Other messaging services such as those operating under a POP3 protocol may require users to configure their accounts on engine 50 to locate the particular server where that messaging service resides. For example, users may

provide extranet addresses with appropriate information to access the email accounts through the messaging services' firewall. The users may also provide a phone number to connect to the messaging service.

In step 420, engine 50 fetches messages from the first messaging service 110 that are new. A message is considered new if engine 50 had not previously retrieved that message. The engine 50 retrieves the message from the first messaging service 110, without delete not the right word altering the content or manner that the message is stored on the first messaging service. The first messaging service 110 operates under a first protocol, such as HTML or POP3 based protocols. Further details on how emails are fetched under these specific protocols are provided with FIGS. 5 and 6.

In step 430, engine 50 fetches messages from the second messaging service 120. The second messaging service 120 operates under a second protocol, such as HTML or POP3 protocols. Embodiments of the invention enable engine 50 to retrieve emails from specified accounts on first and second messaging services even if the first protocol is different from the second protocol. Specifically, as shown by FIG. 3, first messaging service 110 can be a web-based messaging service operating under an HTML protocol, and the second messaging service 120 can be a messaging service operating under a POP3 protocol.

While FIG. 4 illustrates the given user specifying accounts on two messaging services, the given user may configure engine 50 to access any number of accounts on different messaging services 110-130.

In step 440, engine 50 pushes retrieved messages to end device 70. The end device 70 may correspond to another Internet or network enabled computer system, such as a handheld, desktop, or laptop computers. As will be described in greater detail, end device 70 may also correspond to wireless devices such as cell phones, pagers, and WAP enabled devices, such as Sprint PCS type cell-
5 phones.

FIG. 5 illustrates a flow process for fetching an email from a web-based messaging service, such as first messaging service 110. In step 510, engine 50 programmatically accesses configurations for the given user for the web-based
10 messaging service 110. The configurations of the given user identify the specific web-based messaging service 110, as well as the login and password information of the user. The engine 50 includes a specific address to access the given user's account from the messaging service 110, as well as sequence to programmatically enter the user's login and password. The configurations may
15 include other preferences, raw data and directives. For example, the user's configurations may specify the frequency in which engine 50 should access and check first messaging service 110 for messages.

In step 520, engine 50 accesses the web-based messaging service identified by the user's configurations. The user's configuration may identify a
20 specific web address or domain where the user's email account is located. The engine 50 identifies or determines the specific address within the domain that accesses the user's email account. The first messaging service 110 may prompt

00520-155750

engine 50 for information such as login and password of the given user when messaging service 110 is accessed.

In step 530, engine 50 opens the target web page that hosts the user's email account. To open the target web-page, engine 50 signals login and
5 password information to messaging service 110. The login and password information may be signaled in response to prompts from the first messaging service 110 for this information.

Typically, web-based messaging services provide emails as web-based events. The emails are accessible from links on the web page hosting the user's
10 email account. The web page hosting the user's email account also includes links to other web events. For example, the web page may include links to other portal sites, banner ads, and emails. In step 540, engine 50 parses the target web site for links to web events corresponding only to email messages.

Each email message in a messaging service contains a unique signature.
15 The engine 50 may maintain for each user's account signatures of emails already received and/or retrieved. In step 550, engine 50 uses a signature of an email listed as received by the messaging service to determine whether the message is new. The signature of each message may be compared with signatures of emails previously retrieved from first messaging service 110. If in
20 step 555, a determination is made that the message is not new, the message is ignored in step 560. Then in step 565, engine 50 makes a determination to determine whether another message email exists in the email account.

03/22/2014 11:54:47 AM

If an email is determined as being new in step 555, then in step 570 engine 50 retrieves all or part of the new message without deleting the message from the messaging service. The portion of the email may be spliced actively (copied) or passively (undeleted). In either case, the email is left with the messaging service in an unread, or new state. Thus, engine 50 splices the new message from the respective messaging service. This allows the user to access the email account on the messaging service to view new emails, without the account being disturbed by engine 50.

In an embodiment, the user can configure engine 50 to retrieve only the header of the email, or a portion of the email containing header and subject, as well as portions of the body of the text. The portion of the message 50 may be spliced from the original email stored by the messaging service. Once the message is retrieved, engine 50 repeats step 565. Once all emails in the email account have been checked, the process is completed.

FIG. 6 illustrates a flow process for fetching an email from a messaging service operating under a non-web protocol. In an embodiment specified by FIG. 6, engine 50 is assumed to retrieve emails from second messaging service 120, operating under the POP3 protocol.

As with embodiments such as described with FIG. 5, engine 50 in step 610 accesses the user account to identify a non-web messaging service specified by the user. The user account may be configured to provide information such as login and password information. Messaging services operating under a POP3 protocol may operate with systems requiring multiple password and login

information. For example, the messaging service may provide Microsoft Outlook™ or Lotus Cc:mail™, on a Windows NT™ platform. Separate login and password information may have to be provided for the email application and the operating system. Still further, the given user may specify information
5 such as phone numbers to connect to the network containing the messaging service 120.

In step 620, messaging service 120 is accessed by engine 50. Engine 50 may access messaging service 120 through a network connection. In step 630, engine 50 performs a handshaking routine with messaging service 120. Then,
10 engine 50 signals user information such as login and password information to the second messaging service 120 in step 640.

In step 650, each email in the email account maintained by the second messaging application is checked for a signature. If from the signature the determination is made that the email is old, then in step 660, the message is
15 ignored. In step 665, a determination is made as to whether another email message remains to be checked. If the determination is positive, step 650 is repeated. Else, the process is done.

If in step 655, the signature is determined to be new, then in step 670, engine 50 retrieves a portion of the email residing on the second messaging
20 service 120. In an embodiment, engine 50 retrieves the email without deleting the email from the second messaging service 120. For each new email, engine 50 may retrieve the heading, heading and subject line, and/or portions of the body of the email by splicing the portion of the email from the message existing

on the second messaging service 120. This may be accomplished through, for example, executing a <TOP> command using a module of engine 50. Once the portion of the email is retrieved, engine 50 performs step 665 to determine whether another new email exists.

5 C. Retrieving Web-Events for a Wireless End Device

Under an embodiment, engine 50 pushes retrieved network events to a device operating under a protocol different from one or all of sites 10-30. In one embodiment, end device 70 is a WAP device. Engine 50 then retrieves the network events from sites 10-30, reformats the network events for wireless
10 device 70, then pushes the web-events to end device 70.

FIG. 7 illustrates engine 50 in communication with different types of end devices. The user may specify to engine 50 specific end devices 70. For example, the user may access engine 50 through the Internet (via configuration terminal 60) to specify the make, model, or type of end device 70 being used.
15 The engine 50 may access a look-up table or database to determine parameters for converting retrieved messages and other network events to the format of the specified end device 70.

As with previous embodiments, engine 50 communicates with a plurality of network sites 210, 220, and 230 to retrieve network events. In an
20 embodiment, network events include emails and other types of electronic messages. However, other types of network events are also provided for by this embodiment. An example of network events retrievable by a system such as

shown by FIG. 7 includes web events, such as emails and other types of content accessible from an HTML link on a web page.

In an embodiment, engine 50 communicates with a WAP enabled device 272. The engine 50 reformats the network events into HDML after retrieving the network events from network sites 210-230. Then, engine 50 pushes formatted network events to a wireless network. For example, the network event maybe pushed to an uplink server, which then signals the network event to WAP enabled device 272. An example of an end device for this embodiment includes Sprint PCS™ wireless phones. The manner in which the retrieved network events are formatted by engine 50 are configurable by the user through, for example, configuration terminal 60 (FIG. 1).

In an embodiment, engine 50 signals retrieved network events to a pager 274. The engine 50 reformats the retrieved network events into a text format for pagers and similar devices. The pager 274 may have a limited display for alpha-numeric information. The engine 50 accommodates the display format of the pager 274, through, for example, default settings designated to pagers or specific types of pagers. A user may also operate configuration terminal 60 to reconfigure the display format. The engine 50 reformats retrieved network events, reconfigures the retrieved events as specified by the user, and pushes the network events to the pager 274.

In another embodiment, engine 50 signals retrieved network events to a wireless handheld computer 276, such as a Palm VII™. The engine 50 reformat retrieved network events into HDML, or other wireless format for the handheld

computer 276. The network events are then pushed by engine 50 in the HDML format to the end device 276. The user may operate configuration terminal 60 to reconfigure the format used to display or otherwise receive the retrieved network events.

5 Still further, engine 50 may pass the retrieved network events to other computer systems, without reformatting the retrieved network events. For example, end device 70 may be a desktop computer. The engine 50 does not reformat retrieved HTML network events sites 110-130. The engine 50 may, however, reformat retrieved POP3 network events, and reformat those events
10 into an HTML format, before pushing the network events in the HTML format to the end device 276.

In an embodiment, engine 50 is configurable or programmable to reformat network events in any IP format such as HTML or POP3, for any type of end device 70. For example, the user may configure the associated account
15 on engine 50 to provide for a laptop computer, cell-phone and pager as end devices. The engine 50 then retrieves network events, and notifies one or all end devices specified in the user's account.

FIG. 8 illustrates a flow process for a system such as shown and described with FIG. 7. The system may be used to retrieve emails and other
20 electronic messages from messaging services 110-130 (see FIG. 3). The system then reformats the emails into HDML, and pushes the messages to end device 70.

In step 810, a user configures engine 50 to retrieve messages from different electronic messaging services. In step 820, engine 50 retrieves an email from the first messaging service 210. The first messaging service 210 may operate under a first protocol, such as a web-based protocol. In an
5 embodiment, the first messaging service 210 operates under an HTML protocol.

In step 830, engine 50 retrieves a message from a second messaging service 220. The second messaging service 220 may operate under a second protocol. In an embodiment, the second protocol corresponds to POP3.

In step 840, engine 50 reformats the retrieved emails from the first and
10 second protocols to a wireless protocol for the WAP enabled end device 70. An example of a protocol for end device 70 includes HDML. A system for converting HTML and other IP protocols into HDML is provided in provisional U.S. App. No. 60/163,115, incorporated by reference in this application.

In step 850, engine 50 accesses the given user's account to reconfigure
15 the retrieved network events according to configurations specified by the user. The given user can configure the associated account using configuration terminal 60, which accesses their account via the Internet. For example, the user may specify a specific display format, such as the number of lines of text to be displayed from each network event. In email applications, the given user may
20 configure engine 50 to delete selected emails, such as spam.

In step 860, engine 50 pushes the reformatted messages to the end device 70. In an embodiment, the reformatted messages are pushed to a wireless network specified for the end user's device. For example, engine 50

pushes the reformatted messages to a Sprint network if the end device 70 is a Sprint PCS cell-phone.

D. Response System

The given user can respond to retrieved web-events using end device 70.

- 5 In an embodiment, end device 70 is a wireless device, such as a WAP enabled device. The engine 50 notifies engine 50 of retrieved events. In an embodiment, retrieved events, or portions thereof, are pushed to the end device 70. Alphanumeric entries may be entered onto end device 70 in response to notifications of retrieved events. The engine 50 signals the responses to the site
- 10 in which the retrieved events originated.

- As an example, engine 50 may retrieve a web-event corresponding to auction status, including last bid information and reserve amount. This information is signaled to end device 70. The user views the information, and inputs a numerical amount indicating a bid for the item. The engine 50 signals
- 15 the bid amount to the auction site. In this example, engine 50 is pre-programmed to include modules specific to the auction site. The engine 50 knows how to enter login and password information, as well as bid amounts. Furthermore, engine 50 knows how to extract events or contents from web-page through search of key terms or through parsement for links. The user may enter
- 20 configuration information indicating to engine 50 which items or links to items on the auction site to access. The user may also configure engine 50 with payment information, such as credit cards, as well as shipping and billing

addresses. This enables engine 50 to programmatically enter the information into the auction site upon receiving a prompt.

FIG. 9 is a flow process illustrating engine 50 receiving and addressing responses to emails retrieved from messaging services 110-130. An architecture such as described with FIGS. 1 and 3 may be employed with an embodiment such as described with FIG. 9.

In step 910, engine 50 retrieves a new email from one of the messaging services. The email may be retrieved in a manner such as described above. The retrieved email contains header information indicated that the recipient is the user of the end user. The header information also specifies the address of the recipient. The email also contains header information indicating the sender and address of the retrieved email.

In step 920, the email is reformatted for the end device 70. For example, the email may be retrieved from first messaging service 110 (FIG. 3) operating under an HTML protocol, and from second messaging service 120 (FIG. 3) operating under a POP3 protocol. If end device 70 is, for example, a WAP enabled device, then the emails are converted from either HTML or POP3 protocol to an HDML protocol. The email may also be formatted according to configurations specified by the user. The header information, including sender and recipient address of the email, are stored by engine 50.

In step 930, the email is signaled to the end device 70. All or portions of the email may be signaled to the end device. The end device 70 includes a user-interface that enables the recipient to respond to the email. The user may

compose the body of the response using alphanumeric entries, such as through a key pad. The user may also compose the email as a voice file, by for example, speaking into the end device 70.

In step 940, engine 50 receives the response. The engine 50 identifies
5 the response with header information stored in step 920.

In step 950, engine 50 reconfigures header information stored in step 920 and attaches the reconfigured header information to the outgoing response. Thus, the response is provided header information indicating the recipient and recipient address. The recipient and recipient address in the response are
10 determined from header information about the sender of the original email. The response is also provided header information indicating the sender address of the response. The sender address corresponds to the user's email account on the first messaging service.

In step 960, engine 50 reformats the response from the protocol of the
15 end device to a protocol for use with the messaging service of the sender of the original email. For example, engine 50 may convert the response from an HDML protocol used by the WAP enabled device to an HTML protocol. The response is transmitted in step 970 to the sender of the original email. The engine 50 may route the response to the messaging service of the sender
20 through a network such as the Internet.

E. Alternative Embodiments

FIG. 10 illustrates an alternative embodiment in which an end device 470 is notified of web-events occurring on a plurality of sites 410-430. In this

embodiment, configuration terminal 60 communicates via the Internet with the plurality of network sites 410-430. Each of the network sites include engine modules 452-456. The engine modules are programmed to be specific to each associated network site 410-430. In combination, network sites 410-430
5 provide the functions of engine 50, described with previous embodiments.

The engine modules 452-456 are preprogrammed to access network events on sites 410-430. The engine modules 452-456 then reformat the network events for a protocol or medium of end device. In an embodiment, the engine modules 452-456 format the network events for a WAP enabled device,
10 operating under an HDML, WML, or other wireless protocol.

The configuration terminal 60 is used to provide configuration parameters for each site 410-430. The configuration parameters include user input data, such as preferences, directives and other information. The configuration terminal 60 causes engines 452-456 to retrieve network events
15 from each site 410-430 respectively. The information is retrieved by each engine 452-456 according to configurations provided by configuration terminal 60. The user may specify configurations using configuration terminal 60 on an ongoing bases.

In one embodiment, engines 452-456 retrieve emails from sites 410-430.
20 The emails are retrieved according to configurations specified by the user operating the configuration terminal over the Internet.

[illegible]

5